Serial No. 09/456,894

REMARKS

The Applicant and the undersigned thank Examiner El Chanti for his consideration given during the telephonic interview of September 15, 2005 and his careful review of this application. The Examiner imposed a restriction requirement on January 13, 2005. The Applicants elected with traverse Claims 1-26 and 28-29, while Claims 30, 33-40, 42-53, and 55-57 would stand withdrawn.

Upon entry of this amendment, Claims 1-2, 4-12, 14-26, 28-30, and 33-40, 42-53, and 55-57 remain pending in this application. Claims 1-2, 4-12, 14-26 and 28-29 are being prosecuted while Claims 30, 33-40, 42-53, and 55-57 are withdrawn from consideration by the Examiner. Claims 3, 13, 27, 31-32, 41, 54, and 58 have been cancelled. The independent claims are Claims 1, 11, 18, 23, 26, 28, 30, 34, 38, 42, 45, 48, 51, and 55. Consideration of the present application is respectfully requested in light of the above amendments to the application and in view of the following remarks.

Summary of Telephonic Interview of September 15, 2005 - Pursuant to M.P.E.P. § 713.04 and 37 C.F.R. § 1.133(b)

The Applicant and the undersigned appreciate the time and consideration given by Examiner El Chanti during the telephonic interview of September 15, 2005. During this telephonic interview, a proposed claim amendment was presented and discussed. Applicant's representative emphasized that the Examiner's primary prior art reference of U.S. Patent No. 6,314,525 issued in the name of Mahalingham et al. (hereinafter the "Mahalingham reference") does not provide any teaching of the computer code level operations that are described in the amended independent claims. It was explained that the Mahalingham reference describes the disabling of hardware, and specifically, network interface controllers (NICs) 18, 20, and 22 as illustrated in Figure 1 of the Mahalingham reference.

The Examiner and Applicant's representative reviewed the proposed independent claims that describe disabling access to a first section of code in a network driver software interface by overwriting code that is executed before the first section of code with blocking code, while a

21

processor is executing the network driver software interface. The Examiner and the Applicant's representative compared this claim element to the section of the Mahalingham reference that describes initialization of LAN drivers in column 8, lines 41-54.

Applicant's representative did not understand how the section on the initialization of LAN drivers of the Mahalingham reference addressed the claim element of disabling access to a first section of code. The Applicant's representative also stated that the disabling of access to the first section code, as claimed, is performed so that the first section of code can be modified and so that this first section of code (that has been modified) can be later accessed by a processor running the network driver software interface. This claimed subject matter is supported by Figure 12 and pages 20-21 of the originally filed application.

At the end of the telephonic interview, Examiner El Chanti explained that he would investigate these issues further. Examiner El Chanti suggested that the Applicant amend the claims so it is clear that the disabling of access of code is performed so that the modified code can be later accessed or used by the processor. The Applicant notes that the claims as presented in this paper reflect additional changes that are commensurate with the Examiner's helpful comments.

Consideration and approval of this interview summary by Examiner El Chanti are respectfully requested. The Examiner is requested to initial this summary if he approves of it pursuant to M.P.E.P. § 713.04.

Restriction Requirement

The Examiner imposed a restriction requirement on January 13, 2005. The Applicant elected with traverse Claims 1-26 and 28-29, while Claims 30, 33-40, 42-53, and 55-57 would stand withdrawn as non-elected claims. The Examiner has made the restriction requirement final. The Applicant acknowledges the finality of the restriction requirement and that the Applicant has preserved his right to petition the requirement in view of the traversal.

Rejections under 35 U.S.C. § 102(e)

The Examiner rejected Claims 1-26 and 28-29 under 35 U.S.C. 102(e) as being anticipated by the Mahalingham reference. The Applicant respectfully offers remarks to traverse these rejections. The Applicant will address each independent claim separately as the Applicant believes that each independent claim is separately patentable over the prior art of record.

Independent Claim 1

The rejection of Claim 1 is respectfully traversed. It is respectfully submitted that the Mahalingham reference fails to describe, teach, or suggest the combination of: (1) disabling access to at least a first section of computer code in a network driver software interface that is being executed by the processor by (2) overwriting computer code that is executed before the first section of computer code with blocking computer code, (3) wherein the network driver software interface provides for communication between one or more media access control units and one or more protocol drivers in a computer system according to a set of bindings; (4) executing the blocking computer code with the processor; (5) patching the first section of computer code while the blocking computer code of the network driver software interface is being executed by the processor, (6) the patching of the first section of code comprising (7) inserting a template jump from the network driver software interface to a template in a rerouting driver in order to cause the insertion of a rerouting driver into the one or more communication paths provided by the set of bindings; and (8) re-enabling access to the patched first section of computer code by (9) replacing the blocking computer code with computer code that allows execution of the patched first section of computer code, as recited in amended independent Claim 1.

The Mahalingham Reference

The Mahalingham reference describes a server computer 10 illustrated in Figure 1 that is linked through a network backbone 12 to client computers 14 and 16. The server computer 10 includes a group of network interface cards (NICs) 18, 20, 22 which provide communications between the server computer 10 and the network backbone 12. Similarly, the client computer 14

23

includes a network interface card 24 and the client computer 16 includes a network interface card 26 for communicating with the network backbone 12. The network backbone 12 may be a cable such as a 10B2 Thin Ethernet cable, an Ethernet 10BT workgroup hub such as a 3Com Hub 8/TPC, or several interconnected switches or routers 28, 30, 32 such as a Cisco Catalyst 500, as shown in Figure 1.

FIGURE 1
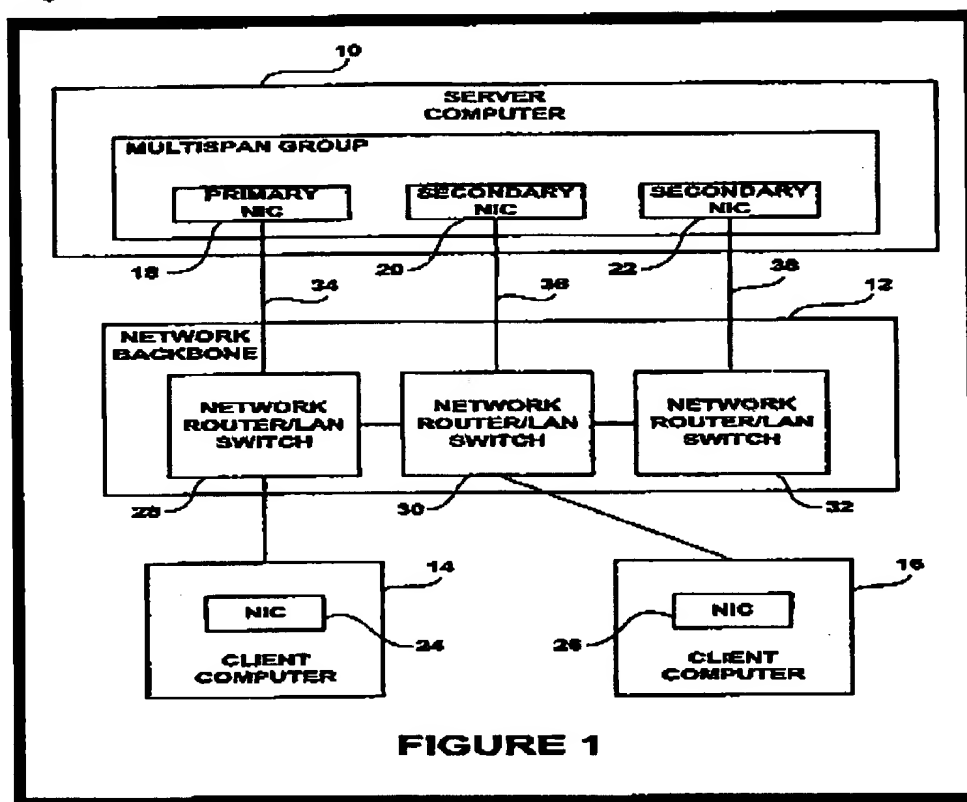
The client computers 14 and 16 make requests for information from the server computer 10 through the network backbone 12. The requests made by the client computers are acknowledged through the primary network interface card (NIC) 18 to the server computer 10. However, if the primary network interface card (NIC) 18, or cable 34 or switch or router 28 fails, the system provides a mechanism for routing network requests through one of the secondary network interface cards (NICs) 20 or 22. The re-routing of network requests is transparent to the client computer 14 or 16. See the Mahalingham reference, column 5, lines 21-35.

24

To address disabling access to at least a first section of computer code in a network driver software interface that is being executed by a processor as recited in Claim 1, the Examiner relies upon the failure of a NIC, such as NIC 18 in Figure 1. See the Examiner's Office Action of April 20, 2005, page 3, lines 13-14 "...the NIC is disabled..."). However, the Applicant respectfully submits that the Mahalingham reference does not provide disabling access to at least a first section of code by overwriting computer code that is executed before the first section of computer code with blocking computer code, as recited in amended independent Claim 1.

The Applicant respectfully submits that the Mahalingham reference provides a teaching of a computer module (the Multispan prescan module) 110 that monitors network interface hardware, NICs 18, 20, and 22 for failures. The computer module 110 can re-route network traffic sent to failed hardware (NIC) to working hardware (NIC):

> "Fault detection and recovery is accomplished by 'MULTISPAN', a process operating within the system. For each group of NICs, if there is a failure virtually on any component related to network traffic, the MULTISPAN process detects the interruption of the data flow and determines which NIC is no longer working. MULTISPAN directs traffic through only the working NICs until the failed NIC is again able to send and receive traffic reliably. Restoring a NIC to reliable operation may involve such steps as replacing a failed NIC (in a computer which supports the hot replacement of failed components), reconnecting or replacing a cable, replacing a failed network switch or router. By placing each NIC in the server on a separate path to the network, MULTISPAN will normally keep the system running until repairs can be accomplished. Being able to schedule repairs decreases cost of owning and operating the computer system." See Mahalingham reference, column 4, lines 7-24.

One of ordinary skill in the art recognizes that this hardware monitoring operation of the Multispan computer module 110 which is separate from any software that could be running on individual NICs 18, 20, 22 in the Mahalingham reference is not equivalent to the code level operations of a single network driver software interface, as described in independent Claim 1.

Therefore, the Mahalingham reference simply does not teach all of the code level elements and operations for a single network driver software interface as recited independent Claim 1. In light of the differences between Claim 1 and the Mahalingham reference mentioned

above, one of ordinary skill in the art recognizes that the Mahalingham reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 1. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Independent Claim 11

The rejection of Claim 11 is respectfully traversed. It is respectfully submitted that the Mahalingham reference fails to describe, teach, or suggest the combination of (1) transmitting from a remote host to a first target computer on a network an installation application and a rerouting driver; (2) transmitting from the remote host to the first target computer a command to cause the first target computer to execute the installation application; (3) the first target computer, responsive to receipt of the command, executing the installation application, wherein the first target computer includes a network driver software interface that provides for communication between one or more media access control units and one or more protocol drivers according to a set of bindings; and (4) the first target computer, responsive to executing the installation application, causing the modification of the network driver software interface to insert the rerouting driver into the one or more communication paths provided by the set of bindings (5) while the network driver software interface is being executed by the first target computer and (6) without restarting the first target computer, (7) the first target computer comprising a multiprocessor system, wherein the insert of the rerouting driver, further comprises: (8) the installation application disabling access to a least a first section of code in the network driver software interface by (9) overwriting code prior to the first section with blocking code; (10) the installation application patching the first section of code while the blocking code is being executed by the processor, the patching comprising (11) inserting a template jump from the network driver software interface to a template in the rerouting driver, as recited in independent Claim 11.

As discussed above with respect to independent Claim 1, the Mahalingham reference does not teach code level changes in a single network driver software interface. Instead, the Mahalingham reference teaches monitoring hardware with a computer module that is external to the hardware.

26

Serial No. 09/456,894

In light of the differences between the claims and the Mahalingham reference mentioned above, one of ordinary skill in the art recognizes that the Mahalingham reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 11. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Independent Claim 18

The rejection of Claim 18 is respectfully traversed. It is respectfully submitted that the Mahalingham reference fails to describe, teach, or suggest a processor for simultaneously executing: (1) a protocol driver; (2) a network driver software interface; (3) a media access control unit; (3) a rerouting driver, (4) wherein during installation of the rerouting driver, a first section of code in the network driver software interface is disabled by (5) overwriting code that is positioned before the first section of code with (6) blocking code, and (7) wherein the first section of code is then patched by (8) inserting a template jump from the network driver software interface to a (9) template in the rerouting driver; (10) the network driver software interface to store a first binding defining a communication path between the protocol driver and the media access control unit, (11) the network driver software interface coupled to communicate packets with the media access control unit, (11) the network driver software interface being patched to communicate the packets with the rerouting driver; and (12) the rerouting driver being executed by the processor at the same time as the network driver software interface and being coupled to communicate the packets with the protocol driver, as recited in amended independent Claim 18.

As discussed above with respect to independent Claim 1, the Mahalingham reference does not teach code level changes in a single network driver software interface. Instead, the Mahalingham reference teaches monitoring hardware with a computer module that is external to the hardware.

In light of the differences between the claims and the Mahalingham reference mentioned above, one of ordinary skill in the art recognizes that the Mahalingham reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 18. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

27

Independent Claim 23

The rejection of Claim 23 is respectfully traversed. It is respectfully submitted that the Mahalingham reference fails to describe, teach, or suggest a rerouting driver comprising: (1) control code, for controlling the rerouting driver; (2) binding code, for establishing at least one binding at the network driver software interface so that (3) the rerouting driver is bound to at least one media access control unit while the network driver software interface and the rerouting driver are executed at the same time; (4) patching code, for inserting template jumps into at least a first section of code in the network driver software interface, (5) the template jumps providing jumps to templates in the rerouting driver (6) so that information from at least one media access control unit destined for at least one protocol driver is rerouted to the rerouting driver while the network driver software interface and the rerouting driver are executed at the same time; (7) at least one template, for receiving information from at least one template jump in the network driver software interface; (8) blocking code, for preventing processing of the patching code that is positioned after the blocking code; and (9) inserted code, for replacing the blocking code and evaluating rerouted information received by the template jumps., as recited in amended independent Claim 23.

As discussed above with respect to independent Claim 1, the Mahalingham reference does not teach code level changes in a single network driver software interface. Instead, the Mahalingham reference teaches monitoring hardware with a computer module that is external to the hardware. And it follows that the Mahalingham reference does not provide any teaching of blocking code or inserted code.

In light of the differences between the claims and the Mahalingham reference mentioned above, one of ordinary skill in the art recognizes that the Mahalingham reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 23. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Serial No. 09/456,894

Independent Claim 26

The rejection of Claim 26 is respectfully traversed. It is respectfully submitted that the Mahalingham reference fails to describe, teach, or suggest the combination of (1) selecting a first section of code of the network driver software interface in a central processing unit that is to be modified while the network driver software interface is running; (2) writing the first section of code of the network driver software interface into the cache memory of the central processing unit while the network driver software interface is running; (3) overwriting a portion of the first section of code in cache memory with blocking code comprising (4) code that causes a loop around serialization instruction in order to create a first version of code while the network driver software interface is running; (5) writing the first version of code into shared memory while the network driver software interface is running; (6) modifying the first version of code in the cache memory to create a second version of code, (7) wherein a portion of the code following the blocking code is overwritten with template jumps to effect a static patch of the network driver software interface when the network driver software interface is running in the shared memory; (8) writing the second version of code into shared memory while the network driver software interface is running; (9) modifying the second version of code in the cache memory with code to create a third version of code, (10) wherein the blocking code is overwritten to remove the blocking code while the network driver software interface is running; and (11) writing the third version of code into shared memory while the network driver software interface is running, as recited in amended independent Claim 26.

As discussed above with respect to independent Claim 1, the Mahalingham reference does not teach code level changes in a single network driver software interface. Instead, the Mahalingham reference teaches monitoring hardware with a computer module that is external to the hardware.

In light of the differences between the claims and the Mahalingham reference mentioned above, one of ordinary skill in the art recognizes that the Mahalingham reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 26. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

29

Independent Claim 28

The rejection of Claim 28 is respectfully traversed. It is respectfully submitted that the Mahalingham reference fails to describe, teach, or suggest the combination of (1) disabling access to a first section of code of a network driver software interface while the network driver software interface is running by (2) overwriting code that is positioned before the first section of code with (3) blocking code, (4) the first section of code providing a communication path between a media access control unit and an application, (5) the first section of code including a generic call; (6) overwriting the first section of code with a second section of code while the network driver software interface is running the blocking code; (7) executing the second section of code to cause execution flow to be rerouted to a third section of code in a rerouting driver, (8) the second section of code being no larger than the first section of code, (9) the third section of code, when executed and while the network driver software interface is running the second section of code, completing the communication path and returning execution flow, the third section of code including additional code not present in the first section of code that is now inserted into the communication path, as recited in amended independent Claim 28.

As discussed above with respect to independent Claim 1, the Mahalingham reference does not teach code level changes in a single network driver software interface. Instead, the Mahalingham reference teaches monitoring hardware with a computer module that is external to the hardware.

In light of the differences between the claims and the Mahalingham reference mentioned above, one of ordinary skill in the art recognizes that the Mahalingham reference cannot anticipate or render obvious the recitations as set forth in amended independent Claim 28. Accordingly, reconsideration and withdrawal of this rejection are respectfully requested.

Dependent Claims 2, 4-10, 12, 14-17, 19-22, 24-26, and 29

The Applicant respectfully submits that the above-identified dependent claims are allowable because the independent claims from which they depend are patentable over the cited references. The Applicant also respectfully submits that the recitations of these dependent claims are of patentable significance.
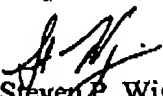
Serial No. 09/456,894

In view of the foregoing, the Applicant respectfully requests that the Examiner withdraw the pending rejections of dependent Claims 2, 4-10, 12, 14-17, 19-22, 24-26, and 29.

CONCLUSION

The foregoing is submitted as a full and complete response to the Office Action mailed on July 20, 2005. The Applicant and the undersigned thank Examiner El-Chanti for consideration of these remarks. The Applicant has amended the claims and has submitted remarks to traverse rejections of Claims 1-26 and 28-29. The Applicant respectfully submits that the present application is in condition for allowance. Such action is hereby courteously solicited.

If the Examiner believes that there are any issues that can be resolved by a telephone conference, or that there are any formalities that can be corrected by an Examiner's amendment, please contact the undersigned in the Atlanta Metropolitan area (404) 572-2884.

Respectfully submitted,

Steven P. Wigmore
Reg. No. 40,447

King & Spalding, LLP
45th Floor
191 Peachtree Street, N.E.
Atlanta, Georgia 30303
404.572.4600
K&S Docket: 05456.105030

31